



Kryptografie

Key Exchange Protokolle

Alexander Fuchs (0106909, 926)

Gerald Haider (0125638, 526)

[Überblick]

- **Teil 1**

- Einführung
- Allgemeines zu Key Exchange Protokollen
- Diffie-Hellman Schlüsselvereinbarung
- TLS (SSL 3.1) Handshake Protokoll

- **Teil 2**

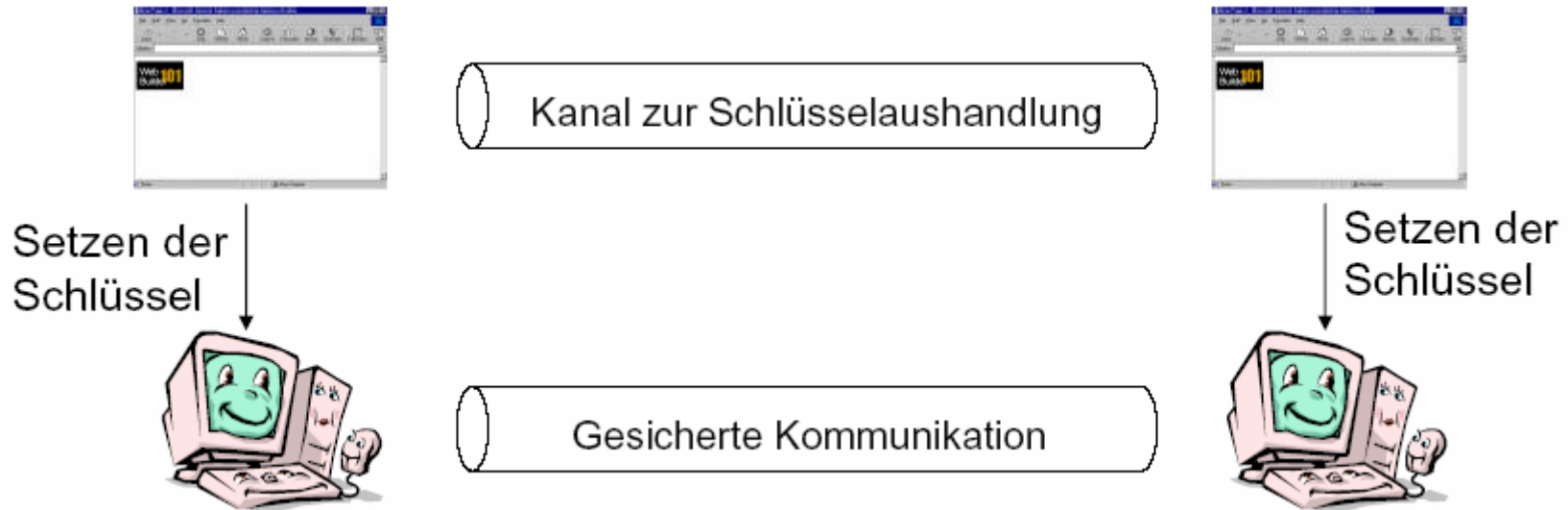
- Internet Key Exchange Protokoll (IKE)
- Alternative Verfahren (Quanten)
- Ausblick und Zusammenfassung

[Einführung]

- **meist symmetrische Verschlüsselung notwendig**
 - kryptografisches Protokoll mit asymmetr. Schlüsseln nicht performant
- **key exchange Problem**
 - tauschen und verwalten von symmetrischen Schlüssel
 - auf vertrauliche Weise
 - früher über gesicherten Kanal oder durch Kurier
 - durch „public/private key“ Verfahren über unsicheren Kanal möglich

[Allgemeines zu Key Exchange Protokolle (1/2)]

- Lassen sich in zwei Kategorien einteilen
 - Verfahren zur Schlüsselvereinbarung (z.B.: Diffie-Hellman)
 - Verfahren zum Schlüsselaustausch (z.B.: RSA)



[Allgemeines zu Key Exchange Protokolle (2/2)]

- **Aushandeln von Kommunikationsparameter**
 - Ver-/Entschlüsselungsverfahren
 - Authentifizierungsverfahren
 - Schlüsselaustauschverfahren

 - **Schlüsselerneuerung notwendig wenn**
 - Lebenszeit des Schlüssels abgelaufen ist
 - Maximal zu schützende Datenmenge gesicherte wurde
 - Neue Attribute für das Schlüsselmaterial benötigt werde
-

Diffie-Hellman Schlüsselvereinbarung (1/5)

- 1976 von Whitfield Diffie und Martin Hellman publiziert
- erlaubt sicheren Key Exchange
 - selbst wenn „Schlüsselgenerierung“ beobachtet wird
- Erzeugung eines gemeinsamen Geheimnisses
- Beruht auf Problematik der Berechnung von
 - diskreten Logarithmen in Restklassen
- löst nicht das „man in the middle“ Problem
 - dazu Sicherstellung der Zugehörigkeit des „public key“ zum Kommunikationspartner notwendig
 - Bsp.: „public-key“ Infrastruktur

[Diffie-Hellman Schlüsselvereinbarung (2/5)]

- Kommunikationspartner einigen sich auf öffentliche Teile
 - Primzahl p
 - Primitivwurzel $g \bmod p$ mit $2 \leq g \leq p-2$
 - Sicherheit wird erhöht wenn g ebenfalls Primzahl
 - Noch besser Sophie-Germain Primzahl
- Beide Kommunikationspartner erzeugen jeweils
 - geheime Zufallszahl a bzw. b aus der Menge $\{0, \dots, p-2\}$
- beide Kommunikationspartner berechnen und übertragen:
 - $A = g^a \bmod p$ (Kommunikationspartner 1)
 - $B = g^b \bmod p$ (Kommunikationspartner 2)
- Anschließend berechnen beide daraus Schlüssel K
 - $K = B^a = (g^b \bmod p)^a$ (Kommunikationspartner 1)
 - $K = A^b = (g^a \bmod p)^b$ (Kommunikationspartner 2)

Diffie-Hellman Schlüsselvereinbarung (2/5)

- Kommunikationspartner einigen sich auf öffentliche Teile
 - Primzahl p
 - Primitivwurzel $g \bmod p$ mit $2 \leq g \leq p-2$
 - Sicherheit wird erhöht wenn g ebenfalls Primzahl
 - Sogenannte Sophie-Germain Primzahl
- beide Kommunikationspartner erzeugen jeweils
 - geheime Zufallszahl a bzw. b aus der Menge $\{0, \dots, p-2\}$
- beide Kommunikationspartner berechnen und übertragen:
 - $A = g^a \bmod p$ (Kommunikationspartner 1)
 - $B = g^b \bmod p$ (Kommunikationspartner 2)
- Anschließend berechnen beide daraus Schlüssel K
 - $K = B^a = (g^b \bmod p)^a$ (Kommunikationspartner 1)
 - $K = A^b = (g^a \bmod p)^b$ (Kommunikationspartner 2)

Diffie-Hellman Schlüsselvereinbarung (2/5)

- Kommunikationspartner einigen sich auf öffentliche Teile
 - Primzahl p
 - Primitivwurzel $g \bmod p$ mit $2 \leq g \leq p-2$
 - Sicherheit wird erhöht wenn g ebenfalls Primzahl
 - Sogenannte Sophie-Germain Primzahl
- beide Kommunikationspartner erzeugen jeweils
 - geheime Zufallszahl a bzw. b aus der Menge $\{0, \dots, p-2\}$
- beide Kommunikationspartner berechnen und übertragen:
 - $A = g^a \bmod p$ (Kommunikationspartner 1)
 - $B = g^b \bmod p$ (Kommunikationspartner 2)
- Anschließend berechnen beide daraus Schlüssel K
 - $K = B^a = (g^b \bmod p)^a$ (Kommunikationspartner 1)
 - $K = A^b = (g^a \bmod p)^b$ (Kommunikationspartner 2)

[Diffie-Hellman Schlüsselvereinbarung (2/5)]

- Kommunikationspartner einigen sich auf öffentliche Teile
 - Primzahl p
 - Primitivwurzel $g \bmod p$ mit $2 \leq g \leq p-2$
 - Sicherheit wird erhöht wenn g ebenfalls Primzahl
 - Sogenannte Sophie-Germain Primzahl
- beide Kommunikationspartner erzeugen jeweils
 - geheime Zufallszahl a bzw. b aus der Menge $\{0, \dots, p-2\}$
- beide Kommunikationspartner berechnen und übertragen:
 - $A = g^a \bmod p$ (Kommunikationspartner 1)
 - $B = g^b \bmod p$ (Kommunikationspartner 2)
- Anschließend berechnen beide daraus Schlüssel K
 - $K = B^a = (g^b \bmod p)^a$ (Kommunikationspartner 1)
 - $K = A^b = (g^a \bmod p)^b$ (Kommunikationspartner 2)

Diffie-Hellman Schlüsselvereinbarung (2/5)

- Kommunikationspartner einigen sich auf öffentliche Teile
 - Primzahl p
 - Primitivwurzel $g \bmod p$ mit $2 \leq g \leq p-2$
 - Sicherheit wird erhöht wenn g ebenfalls Primzahl
 - Sogenannte Sophie-Germain Primzahl
- beide Kommunikationspartner erzeugen jeweils
 - geheime Zufallszahl a bzw. b aus der Menge $\{0, \dots, p-2\}$
- beide Kommunikationspartner berechnen und übertragen:
 - $A = g^a \bmod p$ (Kommunikationspartner 1)
 - $B = g^b \bmod p$ (Kommunikationspartner 2)
- Anschließend berechnen beide daraus Schlüssel K
 - $K = B^a = (g^b \bmod p)^a$ (Kommunikationspartner 1)
 - $K = A^b = (g^a \bmod p)^b$ (Kommunikationspartner 2)

[Diffie-Hellman Schlüsselvereinbarung (3/5)]

- Beispiel
 - **Alice** und **Bob** einigen sich
 - **Primzahl $p = 13$**
 - **Primitivwurzel $g = 2$** ($g \bmod p$ mit $2 \leq g \leq p-2$)
 - Alice und Bob erzeugen jeweils eine geheime Zufallszahl
 - Alice: **$a = 5$**
 - Bob: **$b = 7$**
 - Alice und Bob berechnen und übertragen:
 - Alice: **$A = 2^5 \bmod 13 = 6$** ($g^a \bmod p$)
 - Bob: **$B = 2^7 \bmod 13 = 11$** ($g^b \bmod p$)
 - Anschließend berechnen beide daraus Schlüssel K durch
 - Alice: **$K = 11^5 \bmod 13 = 7$** ($B^a = (g^b)^a$)
 - Bob: **$K = 6^7 \bmod 13 = 7$** ($A^b = (g^a)^b$)
 - beide erhalten den selben Wert, da gilt
 - $(g^a \bmod p)^b = (g^b \bmod p)^a$ bzw. $(g^a)^b = (g^b)^a$

Diffie-Hellman Schlüsselvereinbarung (3/5)

- Beispiel

- **Alice** und **Bob** einigen sich

- **Primzahl $p = 13$**

- **Primitivwurzel $g = 2$** ($g \bmod p$ mit $2 \leq g \leq p-2$)

- Alice und Bob erzeugen jeweils eine geheime Zufallszahl

- Alice: **$a = 5$**

- Bob: **$b = 7$**

- Alice und Bob berechnen und übertragen:

- Alice: **$A = 2^5 \bmod 13 = 6$** ($g^a \bmod p$)

- Bob: **$B = 2^7 \bmod 13 = 11$** ($g^b \bmod p$)

- Anschließend berechnen beide daraus Schlüssel K durch

- Alice: **$K = 11^5 \bmod 13 = 7$** ($B^a = (g^b)^a$)

- Bob: **$K = 6^7 \bmod 13 = 7$** ($A^b = (g^a)^b$)

- beide erhalten den selben Wert, da gilt

- $(g^a \bmod p)^b = (g^b \bmod p)^a$ bzw. $(g^a)^b = (g^b)^a$

Diffie-Hellman Schlüsselvereinbarung (3/5)

- Beispiel
 - **Alice** und **Bob** einigen sich
 - **Primzahl $p = 13$**
 - **Primitivwurzel $g = 2$** ($g \bmod p$ mit $2 \leq g \leq p-2$)
 - Alice und Bob erzeugen jeweils eine geheime Zufallszahl
 - Alice: **$a = 5$**
 - Bob: **$b = 7$**
 - Alice und Bob berechnen und übertragen:
 - Alice: **$A = 2^5 \bmod 13 = 6$** ($g^a \bmod p$)
 - Bob: **$B = 2^7 \bmod 13 = 11$** ($g^b \bmod p$)
 - Anschließend berechnen beide daraus Schlüssel K durch
 - Alice: **$K = 11^5 \bmod 13 = 7$** ($B^a = (g^b)^a$)
 - Bob: **$K = 6^7 \bmod 13 = 7$** ($A^b = (g^a)^b$)
 - beide erhalten den selben Wert, da gilt
 - $(g^a \bmod p)^b = (g^b \bmod p)^a$ bzw. $(g^a)^b = (g^b)^a$

Diffie-Hellman Schlüsselvereinbarung (3/5)

- Beispiel

- **Alice** und **Bob** einigen sich

- **Primzahl $p = 13$**

- **Primitivwurzel $g = 2$** ($g \bmod p$ mit $2 \leq g \leq p-2$)

- Alice und Bob erzeugen jeweils eine geheime Zufallszahl

- Alice: **$a = 5$**

- Bob: **$b = 7$**

- Alice und Bob berechnen und übertragen:

- Alice: **$A = 2^5 \bmod 13 = 6$** ($g^a \bmod p$)

- Bob: **$B = 2^7 \bmod 13 = 11$** ($g^b \bmod p$)

- Anschließend berechnen beide daraus Schlüssel K durch

- Alice: **$K = 11^5 \bmod 13 = 7$** ($B^a = (g^b)^a$)

- Bob: **$K = 6^7 \bmod 13 = 7$** ($A^b = (g^a)^b$)

- beide erhalten den selben Wert, da gilt

- $(g^a \bmod p)^b = (g^b \bmod p)^a$ bzw. $(g^a)^b = (g^b)^a$

Diffie-Hellman Schlüsselvereinbarung (3/5)

- Beispiel
 - **Alice** und **Bob** einigen sich
 - **Primzahl $p = 13$**
 - **Primitivwurzel $g = 2$** ($g \bmod p$ mit $2 \leq g \leq p-2$)
 - Alice und Bob erzeugen jeweils eine geheime Zufallszahl
 - Alice: **$a = 5$**
 - Bob: **$b = 7$**
 - Alice und Bob berechnen und übertragen:
 - Alice: **$A = 2^5 \bmod 13 = 6$** ($g^a \bmod p$)
 - Bob: **$B = 2^7 \bmod 13 = 11$** ($g^b \bmod p$)
 - Anschließend berechnen beide daraus Schlüssel K durch
 - Alice: **$K = 11^5 \bmod 13 = 7$** ($B^a = (g^b)^a$)
 - Bob: **$K = 6^7 \bmod 13 = 7$** ($A^b = (g^a)^b$)
 - beide erhalten den selben Wert, da gilt
 - $(g^a \bmod p)^b = (g^b \bmod p)^a$ bzw. $(g^a)^b = (g^b)^a$

Diffie-Hellman Schlüsselvereinbarung (3/5)

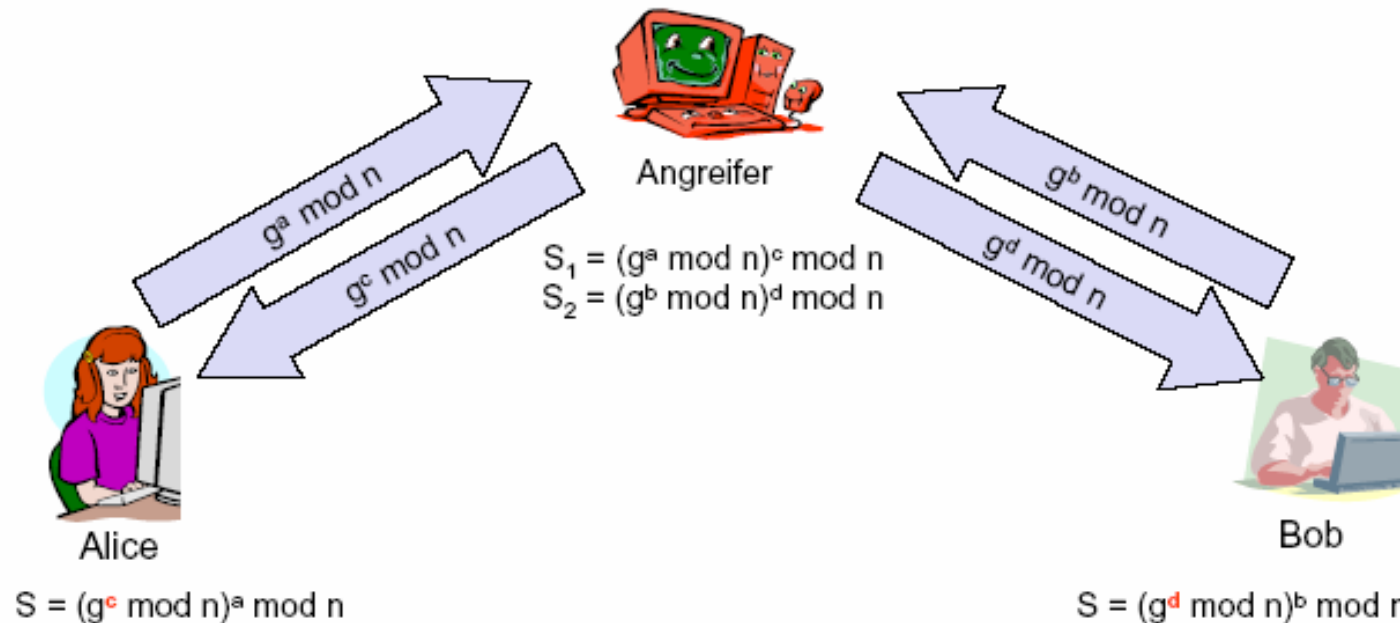
- Beispiel
 - **Alice** und **Bob** einigen sich
 - **Primzahl $p = 13$**
 - **Primitivwurzel $g = 2$** ($g \bmod p$ mit $2 \leq g \leq p-2$)
 - Alice und Bob erzeugen jeweils eine geheime Zufallszahl
 - Alice: **$a = 5$**
 - Bob: **$b = 7$**
 - Alice und Bob berechnen und übertragen:
 - Alice: **$A = 2^5 \bmod 13 = 6$** ($g^a \bmod p$)
 - Bob: **$B = 2^7 \bmod 13 = 11$** ($g^b \bmod p$)
 - Anschließend berechnen beide daraus Schlüssel K durch
 - Alice: **$K = 11^5 \bmod 13 = 7$** ($B^a = (g^b)^a$)
 - Bob: **$K = 6^7 \bmod 13 = 7$** ($A^b = (g^a)^b$)
- beide erhalten den selben Wert, da gilt
 - $(g^a \bmod p)^b = (g^b \bmod p)^a$ bzw. $(g^a)^b = (g^b)^a$

[Diffie-Hellman Schlüsselvereinbarung (4/5)]

- **Sicherheit von DH basiert auf Einwegfunktion**
 - sehr einfach Zahl zu potenzieren
 - Jedoch hoher Aufwand diskreten Logarithmus einer Zahl zu berechnen
 - diskreter Logarithmus in Restklassen
 - es werden sehr große Primzahlen verwendet
 - Diffie Hellman Problem $K = g^{a*b} \bmod p$ lösbar wenn
 - Man diskreten Logarithmus mod p berechnen kann
 - Mit heutigen Mitteln würden Rechner die Lebenszeit eines Universums benötigen wenn:
 - Primzahl mehr als 300 Stellen hat
 - a bzw. b mindestens 100 Stellen lang sind

Diffie-Hellman Schlüsselvereinbarung (5/5)

- „man in the middle“ Problem
 - Angreifer gibt sich Bob gegenüber als Alice aus und umgekehrt
 - fungiert als eine Art Dolmetscher



TLS (SSL 3.1) Handshake Protokoll

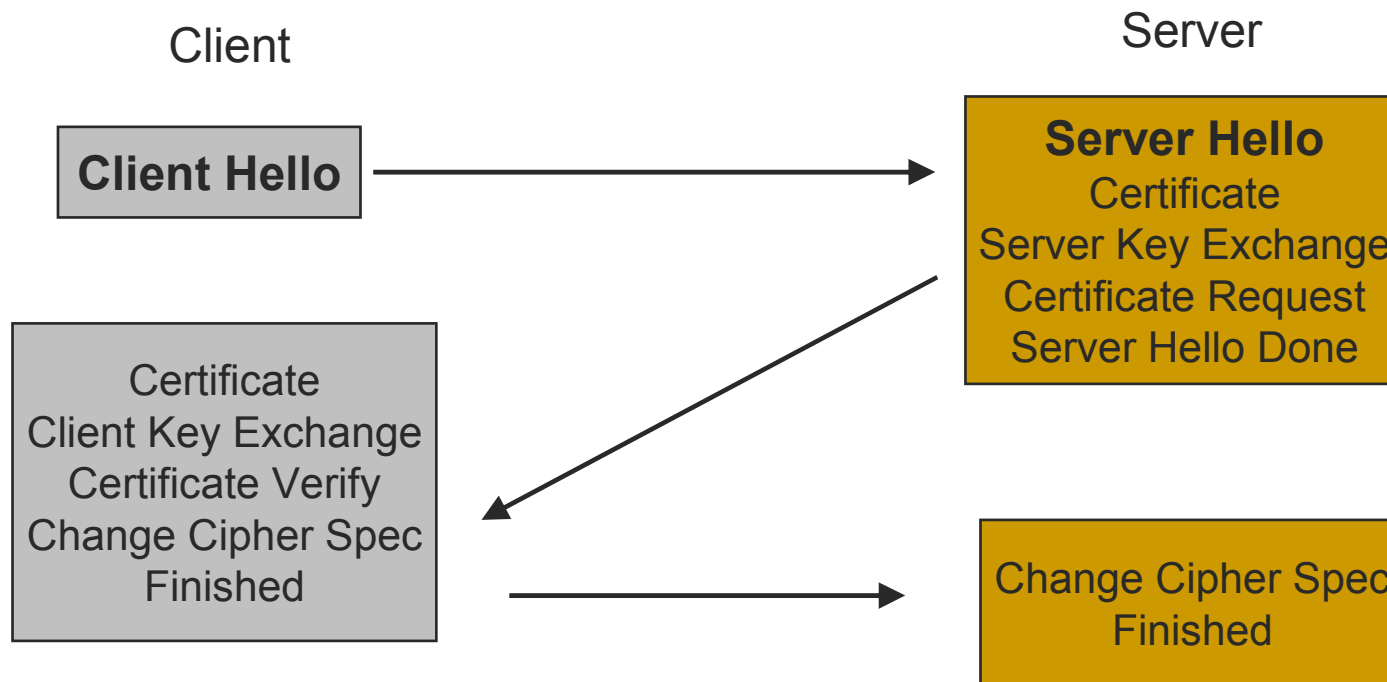
- **Transport Layer Security (TLS)**
 - **Zwischen Transport und Applikationslayer**
 - Wird vom Applikationslayer als Transportlayer
 - Und vom Transportlayer als Applikationslayer wahrgenommen
 - **Verwendung von kryptographische Verfahren**
 - symmetrische Verschlüsselung (DES, RC4, IDEA,..)
 - asymmetrische Verschlüsselung (RSA, DSS,..)
 - One Way Hashfunktionen (MD5,SHA..)
 - Zertifizierung

TLS (SSL 3.1) Handshake Protokoll

- Erlaubt Privacy, Datenintegrität und Authentizität
 - zwischen zwei Anwendungen
- Besteht grundsätzlich aus zwei Protokollen (Schichten)
 - Handshakeprotokoll (Verbindungsaufbau)
 - Rekordprotokoll (verschlüsselte Kommunikation)
- Aufgaben des Handshake Protokolls
 - Aushandeln von Verbindungsmodalitäten (Verschlüsselungsverfahren,...)
 - Austausch von Zertifikaten
 - Schlüsselaustausch

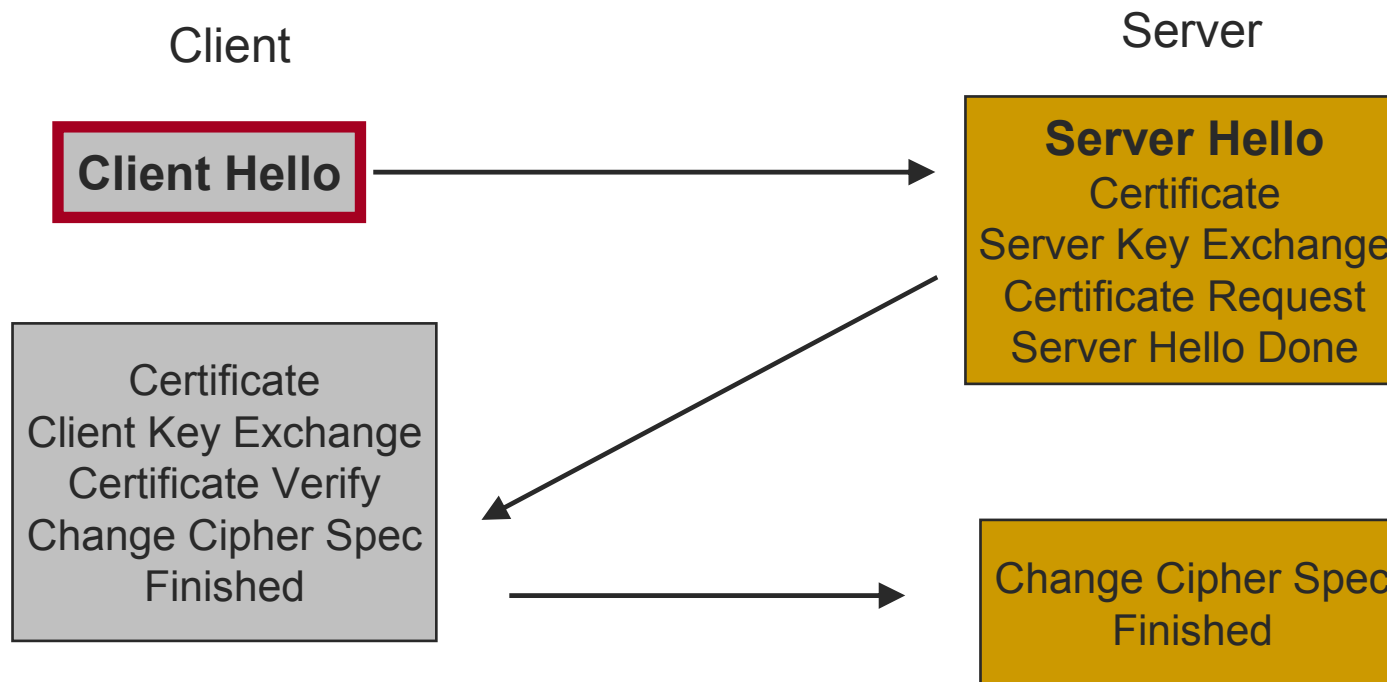
TLS (SSL 3.1) Handshake Protokoll

■ Ablauf des Handshakes



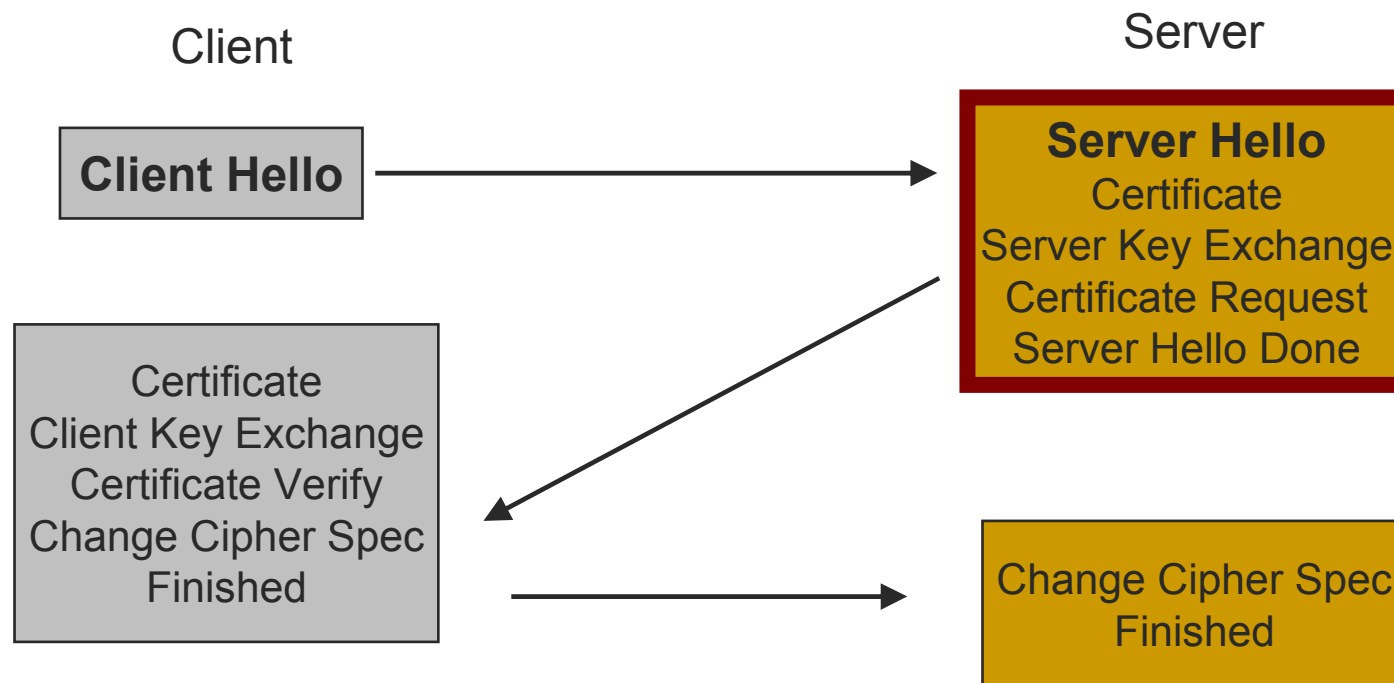
TLS (SSL 3.1) Handshake Protokoll

■ Ablauf des Handshakes



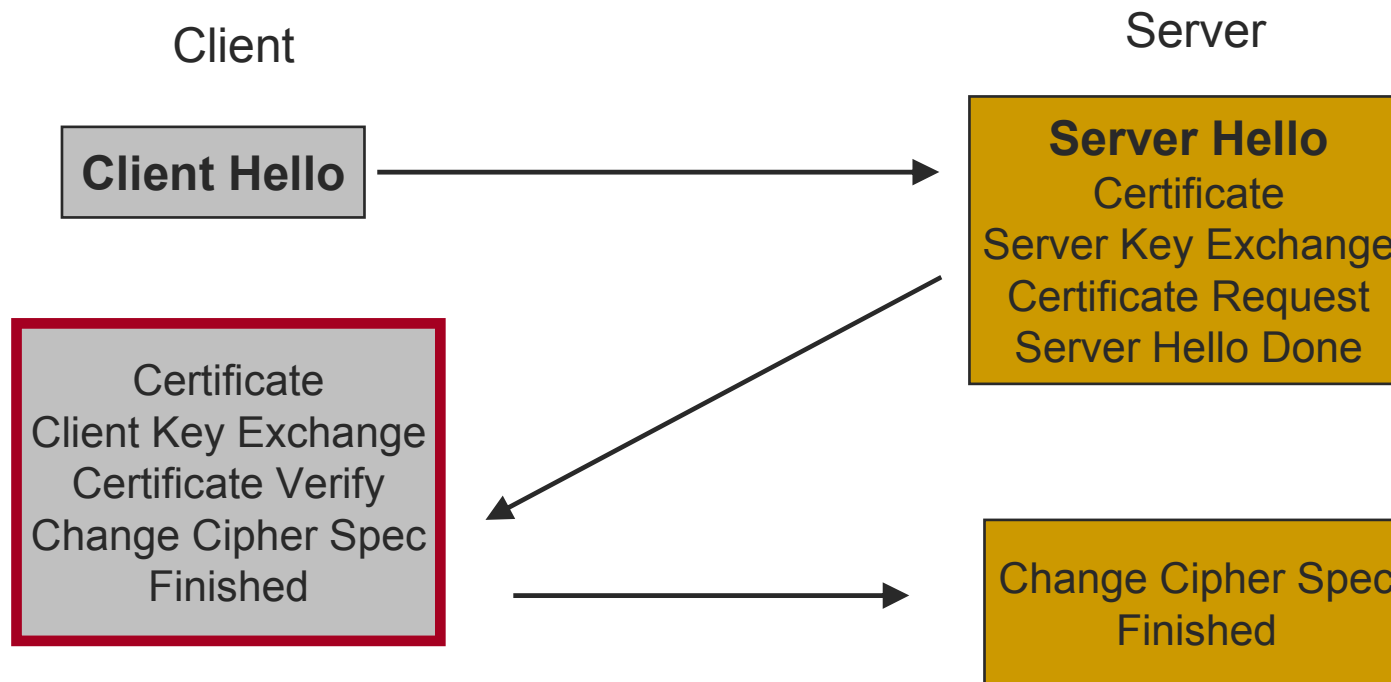
TLS (SSL 3.1) Handshake Protokoll

■ Ablauf des Handshakes



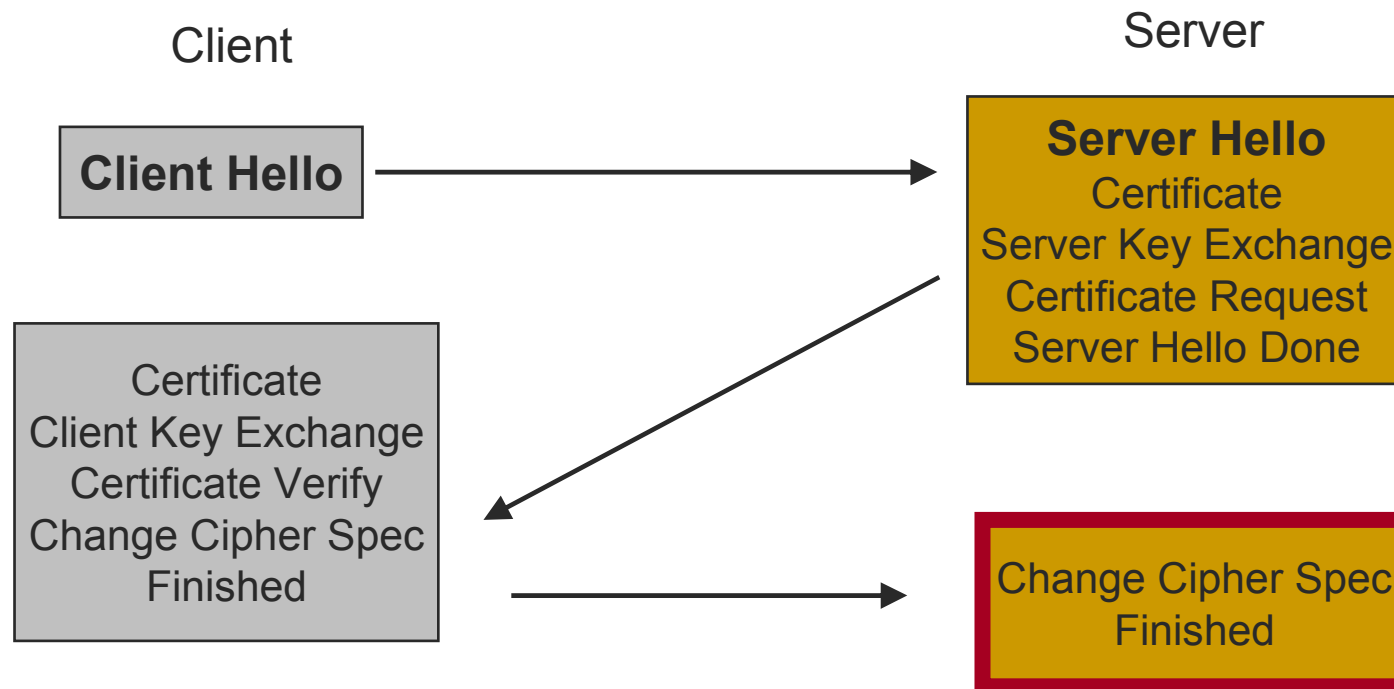
TLS (SSL 3.1) Handshake Protokoll

■ Ablauf des Handshakes



TLS (SSL 3.1) Handshake Protokoll

■ Ablauf des Handshakes



TLS (SSL 3.1) Handshake Protokoll

- Schlüsselaustausch (SSL 2.0 unterstützte nur RSA)
 - **DHE_DSS und DHE_RSA**
 - ServerKey Exchange enthält Diffie-Hellman (DH) Parameter (signiert)
 - ClientKey Exchange enthält DH Anteil des Clients
 - **RSA**
 - Pre Master Secret wird durch Client erzeugt
 - Server Key Exchange entfällt
 - Client Key Exchange für den Server RSA verschlüsselt
 - **RSA_EXPORT**
 - ServerKeyExchange enthält temporären signierten RSA Schlüssel
 - Client Key Exchange mit temp. RSA Schlüssel verschlüsselt

[Internet Key Exchange (IKE)]

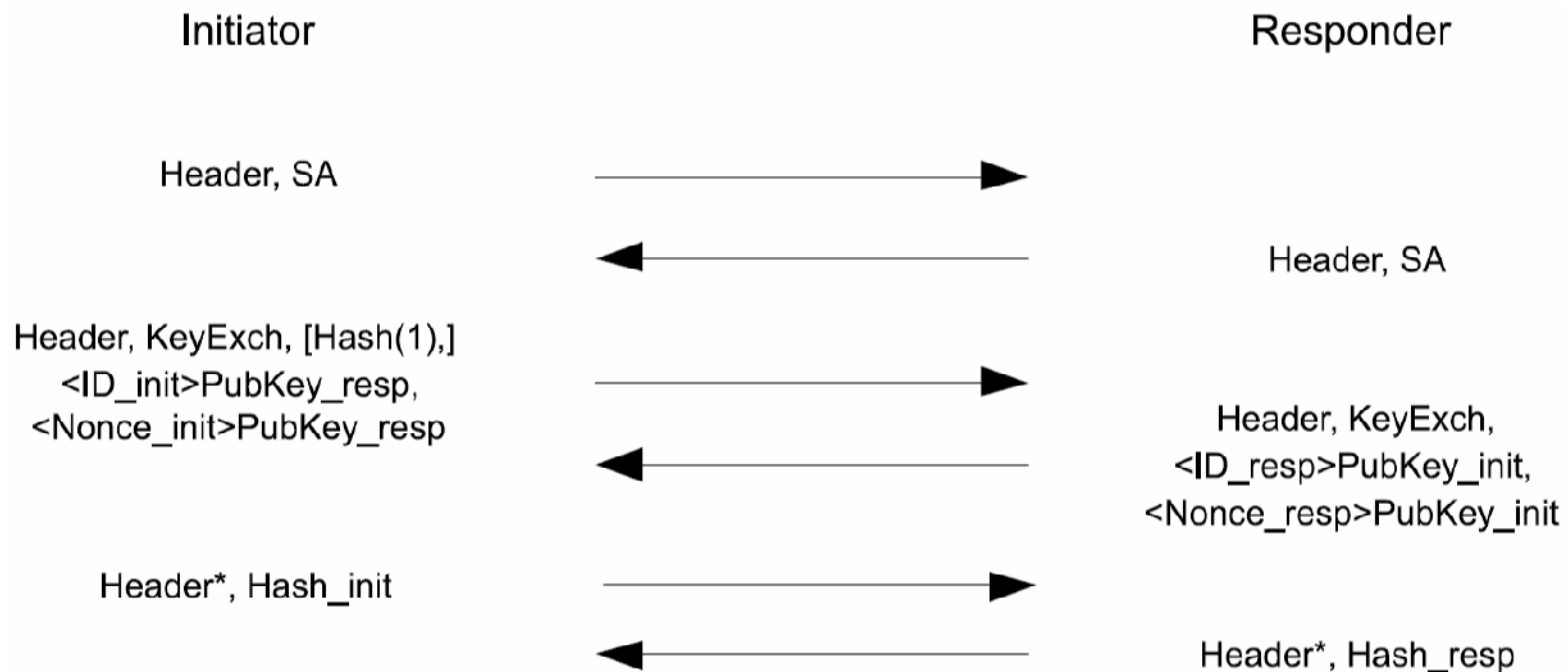
- IKE Protokoll dient der automatischen Schlüsselverwaltung für IPsec
- verwendet DH-Schlüsselaustausch
- IKE ist in [RFC 2409](#) spezifiziert
- IKE basiert auf:
 - Internet Security Association and Key Management Protokoll (ISAKMP, [RFC 2408](#))
 - *Domain of Interpretation* (DOI, [RFC 2407](#))
 - Oakley Key Determination Protocol (OAKLEY, [RFC 2412](#))
 - Secure Key Exchange Mechanism (SKEME)
- Netzwerkschicht (Schicht 3) des OSI-Modells.

[IKE – Phase 1]

- 8 Varianten von Phase 1
 - Type of Keys:
 - pre-shared symmetric key
 - old-style public encryption key
 - new-style public encryption key
 - public signature-verification key
 - Für jeden Schlüsseltypen 2 Austauschmodi:
 - „Main Mode“ – 6 Nachrichten
 - „Aggressive Mode“ – nur 3 Nachrichten

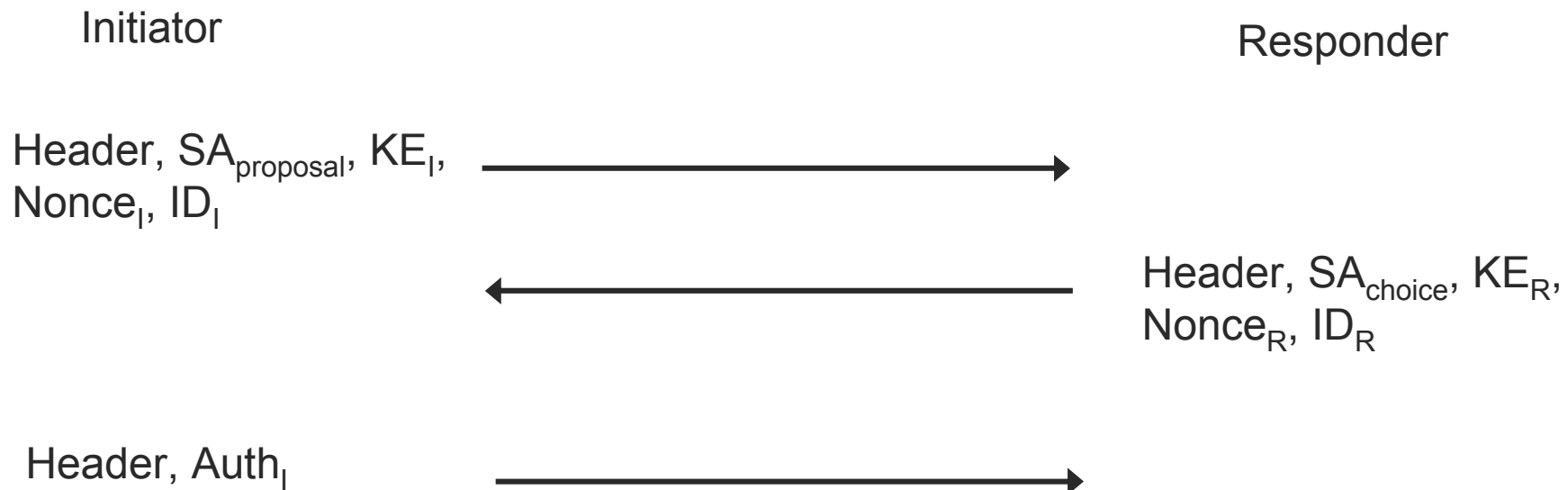
[IKE – Phase 1 – Main Mode]

- Beispiel: Main Mode mit öffentlichen Schlüsseln



IKE – Phase 1 – Aggressiv Mode

- Beispiel: Aggressiv Mode mit öffentlichen Schlüsseln



[IKE – Phase 2]

- nur nach Phase 1
- „Quick Mode“ nutzt den Session Key der in Phase 1 ausgehandelt wurde
- mehrere Phase 2 Exchanges möglich um Verbindungen mit verschiedenen Sicherheitsstufen zu ermöglichen
 - "integrity-only"
 - "confidentiality-only"
 - "encryption with a short key"
 - "encryption with a strong key."
- Ziel: für ESP und AH Anwendungsstrategien zu verhandeln und Schlüsselmaterial zu erzeugen

[Ausblick]

- **Quantenkryptografie:** eigentlich quantenmechanischer Schlüsselaustausch, KEIN Verschlüsselungsverfahren
- Verschlüsselung selbst erfolgt über **One-Time-Pad**

[Zusammenfassung]

- **Teil 1**

- Einführung
- Allgemeines zu Key Exchange Protokollen
- Diffie-Hellman Schlüsselvereinbarung
- TLS (SSL 3.1) Handshake Protokoll

- **Teil 2**

- Internet Key Exchange Protokoll (IKE)
- Alternative Verfahren (Quanten)
- Ausblick und Zusammenfassung

[Referenzen]

- Menezes, Oorschot, Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.
- Wenbo Mao, *Modern Cryptography: Theory and Practice*, Prentice Hall, 2003
- IPsec, Internet: <http://de.wikipedia.org/wiki/IKE> Zugriff: 15.12.2004
- Wolfgang Thomas, *IPSec Architektur und Protokolle, Internet Key Exchange (IKE)*, Internet: <http://www.net.informatik.tu-muenchen.de/teaching/WS02/security/securityUeb/07ausarbeit.pdf>